REMARKS

As an initial matter, Applicant wishes to thank the Examiner for re-opening prosecution. Applicant also wishes to thank the Examiner for withdrawing the claim rejections based on Rys.

Amendments to claims 1 and 15 are for the purpose of clarifying what Applicant regards as the claimed invention. No new matter has been added.

I. Claim Rejections under 35 U.S.C. § 101

Claims 1-25 stand rejected under 35 U.S.C. § 101 because the claims are allegedly directed to a process that is not tied to another statutory class and does not transform the underlying subject matter to a different state or thing. Independent claims 1 and 15 have been amended to recite a "processor," which is well known to include at least some hardware. Thus, the subject matter of claims 1 and 15 are tied to a statutory class. Also, claim 1 recites outputting the result to a data stream, and claim 15 recites outputting an output to a data stream. Applicant respectfully submits that the act of outputting result/output to a data stream results in a physical transformation (because such act results in a data stream having the result/output). For at least the foregoing reasons, claims 1 and 15 should satisfy § 101.

II. Claim Rejections under 35 USC § 103

Claims 1-43 stand rejected under 35 U.S.C. 103(a) as allegedly being unpatentable over "BURG – Fast Optimal Instruction Selection and Tree Parsing" (Fraser) in view of "Better XML Parser through Functional Programming" (Kiselyov), and further in view of US 2002/0188613 (Chakraborty).

A. Cited passages of Fraser do not disclose or suggest an operator tree as described in the claims.

Claim 1 recites *an operator tree associated with a plurality of operators* that correspond with a *program statement* in a database query language (Emphasis Added). Claims 15 and 26-29 recite similar limitations. According to the Office Action, the Overview of Fraser allegedly discloses such an operator tree. However, the Overview of Fraser actually discloses, inter alia:

> BURG is a program that generates a fast tree parser using BURS (Bottom-Up Rewrite System) technology. *It accepts a cost-augmented tree grammar and emits a C program that discovers in linear time an optimal parse of trees in the language described by the grammar.* BURG has been used to construct fast optimal instruction selectors for use in code generation. BURG addresses many of the problems addressed by TWIG [AGT89, App87], but it is somewhat less flexible and much faster. BURG is available via anonymous ftp from kaese.cs.wisc.edu. The compressed shar file pub/burg.shar.Z holds the complete distribution.
>
> (Emphasis Added)

Thus, the cited passage of Fraser actually discloses a program that accepts a tree grammar and outputs a C program that discovers an optimal parse of trees. There is nothing in the cited passage of Fraser that discloses or suggests a plurality of *operators* that are associated with a tree, nor is there anything in the cited passage of Fraser that discloses or suggest that such operators correspond with a *program statement in a database query language*, as recited in the claims.

Kiselyov and Chakraborty also do not disclose or suggest the above limitations, nor are they being relied upon for the disclosure of the above limitations. Since none of the cited references, either alone or in combination, discloses or suggests the above limitations, any purported combination of the cited references cannot result in the subject matter of claims 1, 15, and 26-29. For at least the foregoing reasons, Applicant respectfully submits that the prima facie

case of the § 103 rejections based on Fraser, Kiselyov, Chakraborty has not been established, and requests that the § 103 rejection be withdrawn.

B. Cited passages of Kiselyov do not disclose or suggest outputting the result for a child node to a data stream without buffering the result or an intermediate result in storage when top-down processing is performed.

Claim 1 also recites outputting the result for a *child node* to a data stream without buffering the result or an intermediate result in storage when top-down processing is performed (Emphasis Added). Claims 26 and 27 recite similar limitations. Claim 15 recites that the output from top-down processing the first *child operator node* is output to a data stream without buffering the result or an intermediate result in storage (Emphasis Added). Claims 28 and 29 recite similar limitations. Thus, these claims describe that the result that is not buffered in storage is for *a child node/child operator node*. Applicant agrees with the Examiner on page 5 of the Office Action that Fraser does not disclose such limitations.

According to the Office Action, section 4.2, fifth paragraph, and figure 4 of Kiselyov allegedly disclose the above limitations. However, the cited passage of Kiselyov actually discloses:

> The function ssax-outline also illustrates the benefit of the SAX XML parsing mode. *The function prints element names as they are identified and accumulates no data.* It can therefore process documents of arbitrary size – far bigger than the amount of available memory. The function ssax-outline is a true stream processor, with low memory requirements and low latency.
>
> (Emphasis Added)

Thus, the cited passage of Kiselyov discloses that the function ssax-outline does not accumulate data, but there is noting in the cited passage that discloses or suggests that a result for *a child*

node / child operator node is not buffered in storage. Also, just because data is not "accumulated," it does not mean that data is not "stored" (this is because data may be stored without being accumulated – e.g., by writing over previous data). Notably, the element names printed by the ssax-outline function in Kiselyov are for presentation to a user. Thus, the printed element names in Kiselyov are not for a child node / child operator node.

Chakraborty also does not disclose or suggest the above limitations, nor is it being relied upon for the disclosure of the above limitations. Since none of the cited references, either alone or in combination, discloses or suggests the above limitations, any purported combination of the cited references cannot result in the subject matter of claims 1, 15, and 26-29. For these additional reasons, claims 1, 15, and 26-29, and their respective dependent claims, are believed allowable over Fraser, Kiselyov, Chakraborty, and their combination.

C. Cited references do not disclose or suggest the limitations regarding top-down processing.

Claims 1, 26, and 27

Claim 1 recites "determining if the child node relates to an operator for which top-down processing is capable of being performed, wherein the top-down processing is capable of being performed when a result for the operator is capable of being generated without storage of the result for the parent operator node." Claims 26 and 27 recite similar limitations. Applicants respectfully submit that Fraser, Kiselyov, and Chakraborty do not disclose or suggest the above limitations.

According to page 5 of the Office Action, Section 3, first paragraph of Fraser allegedly discloses determining if the child node relates to an operator for which top-down processing is capable of being performed. Applicant respectfully disagrees. The cited passage of Fraser

actually describes traversing "the subject tree twice", wherein the first pass "runs bottom-up" and the second pass "traverses the subject tree top-down." Thus, in Fraser, each node is always related to a tree than can be traversed top-down. As such, there is no need in Fraser to determine if a child node relates to an operator for which top-down processing is capable of being performed, and Fraser in fact teaches away from such act (as described in claims 1, 26, and 27). Note that a reference that teaches away from the subject matter of the claim cannot be used to establish a prima facie case of a § 103 rejection.

Also, according to page 5 of the Office Action, in Fraser, "the skipping of the subtree means that top-down processing cannot be performed." Applicant respectfully submits that this characterization of Fraser is improper. Rather, Section 3, paragraph 1 of Fraser specifically describes that when a subject node is skipped, "the reducer may proceed directly to grandchildren, great-grandchildren, and so on." Thus, Fraser clearly describes that when a node is skipped, the process may still be continued in a top-down manner – which again, teaches away from the subject matter of the claims. Note that a reference that teaches away from the subject matter of the claim cannot be used to establish a prima facie case of a § 103 rejection.

In addition, Applicant respectfully notes that claim 1 does not merely recite top-down processing, but it actually recites *the act of determining if* the child node relates to an operator for which top-down processing is capable of being performed. Applicant also respectfully notes that each claim element must be considered and given patentable weight, and therefore, the limitation regarding the *act of determining if* a child node relates to an operator cannot be ignored. In this case, there is nothing in Fraser that discloses or suggests any *act of determining if* a child node relates to an operator, nor does Fraser disclose or suggest any *act of determining if* a child node relates to an operator for which top-down processing is capable of being performed.

Also, Kiselyov does not disclose or suggest that the first child operator node is eligible for the top-down processing when a result for an operator associated with the first child operator node is capable of being generated without storage of the result for the parent operator node. In particular, Kiselyov says nothing about the condition when a child operator node is eligible for top-down processing – i.e., the condition being that a result for an operator associated with the first child operator node is capable of being generated without storage of the result for a parent operator node, as described in the claims. Rather, the cited passage (Section 4.2, fifth paragraph) of Kiselyov discloses a function ssax-outline that does not accumulate data. There is nothing in the cited passage of Kiselyov that discloses or suggests that a result for a parent operator node is not *stored* (Note that just because data is not "accumulated," it does not mean that it is not "stored", as discussed). Also, for the sake of argument, even assuming that the non-accumulation of data in Kiselyov is analogized as the claimed "without storage of the result for a parent operator node" (which as discussed, is improper), there is nothing in Kiselyov that discloses or suggests that the feature "a result for the operator is capable of being generated without storage of the result for the parent operator node" is the condition for determining whether the top-down processing is capable of being performed, as described in the claims.

For these additional reasons, claims 1, 26, and 27, and their respective dependent claims, are believed allowable over Fraser, Kiselyov, Chakraborty, and their combination.

Claims 15, 28, and 29

Claim 15 recites determining whether the parent operator node is related to a first child operator node that is eligible for top-down processing, wherein the first child operator node is eligible for the top-down processing when a result for an operator associated with the first child operator node is capable of being generated without storage of the result for the parent operator

node. Claims 28 and 29 recite similar limitations. Applicants respectfully submit that Fraser, Kiselyov, and Chakraborty do not disclose or suggest the above limitations.

According to page 13 of the Office Action, Section 2, fifth paragraph, and Section 3, first paragraph of Fraser allegedly disclose the above limitations. Applicant respectfully disagrees. The cited passage of Fraser actually describes traversing "the subject tree twice", wherein the first pass "runs bottom-up" and the second pass "traverses the subject tree top-down." Notably, in Fraser, the nodes in the tree of Fraser are always presumed to be related. As such, there is no need in Fraser to determine if a parent node relates to a child node that is eligible for top-down processing, and Fraser in fact teaches away from such act (as described in claims 15, 28, and 29). Note that a reference that teaches away from the subject matter of the claim cannot be used to establish a prima facie case of a § 103 rejection.

Also, according to page 13 of the Office Action, in Fraser, "the skipping of the subtree means that top-down processing cannot be performed." Applicant respectfully submits that this characterization of Fraser is improper. Rather, Section 3, paragraph 1 of Fraser specifically describes that when a subject node is skipped, "the reducer may proceed directly to grandchildren, great-grandchildren, and so on." Thus, Fraser clearly describes that when a node is skipped, the process may still be continued in a top-down manner – which again, teaches away from the subject matter of the claims. Note that a reference that teaches away from the subject matter of the claim cannot be used to establish a prima facie case of a § 103 rejection.

In addition, Applicant respectfully notes that claim 15 does not merely recite top-down processing, but it actually recites *the act of determining whether* the parent operator node is related to a first child operator node that is eligible for top-down processing. Applicant also respectfully notes that each claim element must be considered and given patentable weight, and

therefore, the limitation regarding the *act of determining whether* the parent operator node is related to a first child operator node cannot be ignored. In this case, there is nothing in Fraser that discloses or suggests any *act of determining whether* the parent operator node is related to a first child operator node (rather, as discussed, the nodes in the tree of Fraser are presumed to be always related), nor does Fraser discloses or suggests the act of determining whether the parent operator node is related to a child operator node *that is eligible for top-down processing.*

Also, Kiselyov does not disclose or suggest that the first child operator node is eligible for the top-down processing when a result for an operator associated with the first child operator node is capable of being generated without storage of the result for the parent operator node. In particular, Kiselyov says nothing about the condition when a child operator node is eligible for top-down processing – i.e., the condition being that a result for an operator associated with the first child operator node is capable of being generated without storage of the result for a parent operator node. Rather, the cited passage (Section 4.2, fifth paragraph) of Kiselyov discloses a function ssax-outline that does not accumulate data. There is nothing in the cited passage of Kiselyov that discloses or suggests that a result for a parent operator node is not *stored* (Note that just because data is not "accumulated," it does not mean that it is not "stored", as discussed). Also, for the sake of argument, even assuming that the non-accumulation of data in Kiselyov is analogized as the claimed "without storage of the result for a parent operator node" (which as discussed, is improper), there is nothing in Kiselyov that discloses or suggests that the feature "a result for the operator is capable of being generated without storage of the result for the parent operator node" is the condition for determining whether the top-down processing is capable of being performed, as described in the claims.

For these additional reasons, claims 15, 28, and 29, and their respective dependent claims, are believed allowable over Fraser, Kiselyov, Chakraborty, and their combination.

CONCLUSION

If the Examiner has any questions or comments regarding this response, please contact the undersigned at the number listed below.

To the extent that any arguments and disclaimers were presented to distinguish prior art, or for other reasons substantially related to patentability, during the prosecution of any and all parent and related application(s)/patent(s), Applicant(s) hereby explicitly retracts and rescinds any and all such arguments and disclaimers, and respectfully requests that the Examiner re-visit the prior art that such arguments and disclaimers were made to avoid.

The Commissioner is authorized to charge any fees due in connection with the filing of this document to Vista IP Law Group's Deposit Account No. **50-1105**, referencing billing number **OID-2003-207-01**. The Commissioner is authorized to credit any overpayment or to charge any underpayment to Vista IP Law Group's Deposit Account No. **50-1105**, referencing billing number **OID-2003-207-01**.

Respectfully submitted,

DATE: September 9, 2009

By: /Gerald Chan/
Gerald Chan
Registration No. 51,541

VISTA IP LAW GROUP, LLP
1885 Lundy Ave., Suite 108
San Jose, California 95131
Telephone: (408) 321-8663 (Ext. 203)
Facsimile: (408) 877-1662